`Save your work as a jupyter notebook with your group name. The first cell of the jupyter notebook must contain your heading AND the names and index numbers of your group members.`

`You must also format your jupyter notebook so that it can be displayed as a presentation. You can look up how to do this online.`

1. We discussed the modular exponentiation algorithm in class. Convert this into a python function. Test that your function gives the correct answer using the example $3^{644} \bmod 645$ which we discussed in class.

   Now, evaluate

   $7^{121} \bmod 13$. You should do this **three** ways. First, find 7^{121}, and then find the result of that modulo 13. Next, use the function you have written to evaluate result. Finally, use python's built in % function to evaluate the result. Which procedure works fastest? Which is slowest?

2. We are going to explore the complexity of the RSA algorithm by trying to decrypt the system **without knowing** the prime numbers. Given $p = 43$, $q = 59$, and $e = 13$, determine how long it takes to find the decryption key d by brute force. Specifically, $n = pq = 2537$, $e = 13$ and d is an inverse to e modulo $(42 \cdot 58)$.

   Now select

   - two four-digit prime numbers
   - two six-digit prime numbers
   - two eight-digit prime numbers
   - two ten-digit prime numbers

   and using a preselected key e (to make things easier, use a prime number for e), find the time it takes to evaluate d by brute force. You'll need to factor pq first, then work out d as well.

   Plot the times on a graph. If the time differences are large, use a log scale for your graph.

3. Let us explore brute-force password cracking.

   (a) Suppose that a pin code has four digits. Write a python function to determine the code by brute-force method (checking each pin in turn). Test your function with 500 randomly generated digits, and plot the distribution of times it takes to crack the password.

   (b) Repeat the above exercise for six decimal digits.

   (c) Repeat the above exercise if the code can have alphanumeric characters (case sensitive). Do this for both four and six digits.